APU Neural Net Tool (ANNT) Thor Design Panel 3

Document # 84k-01800-020

October 31, 1997

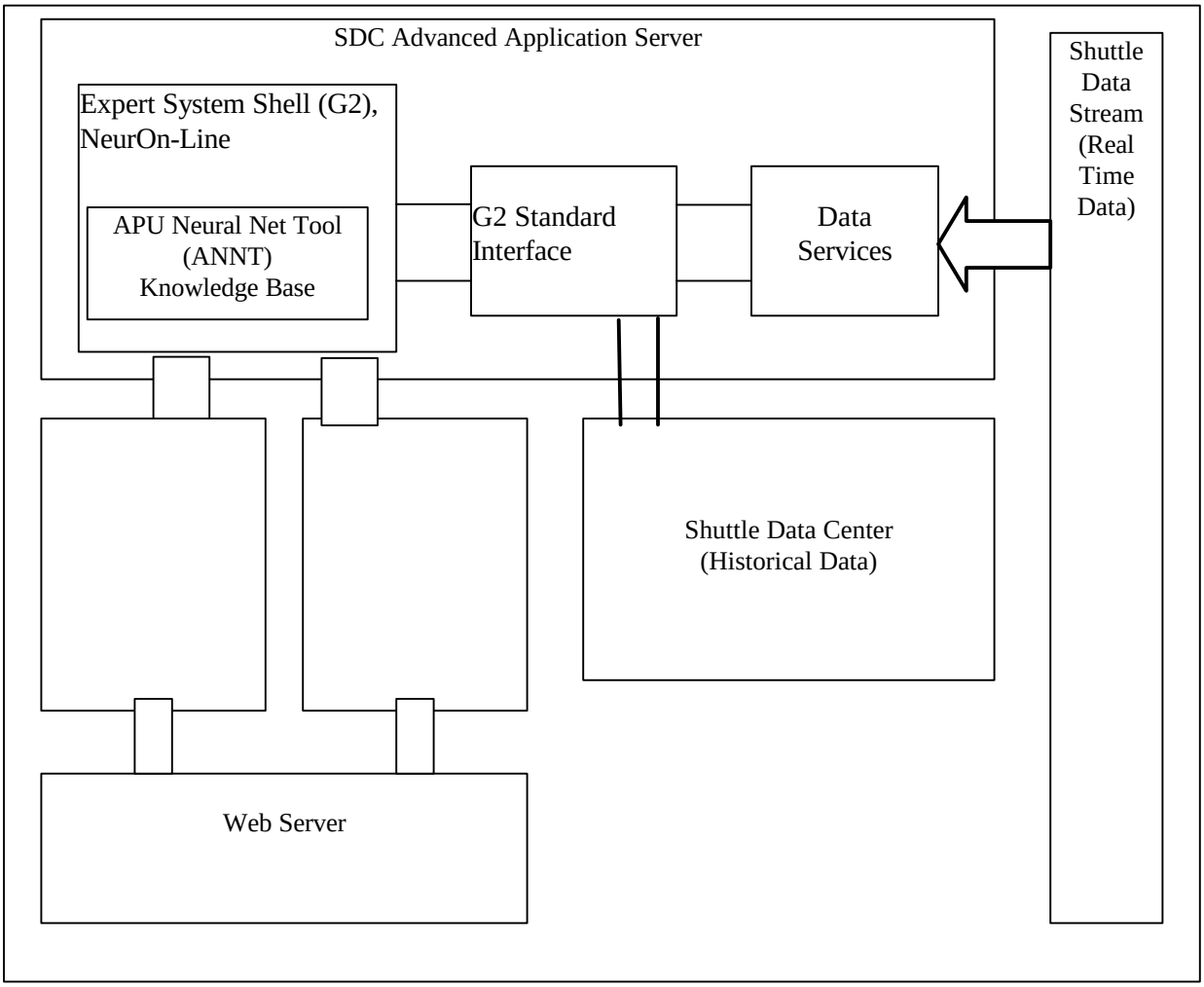
Version 1.0

1. APU Neural Net Tool

1.1 APU Neural Net Tool Introduction

1.1.1 APU Neural Net Tool Overview

The APU Neural Net Tool (ANNT) is an expert system software application that has been developed using Gensym's G2 and NeurOn-Line software. The tool combines a real-time display, individual pulse diagnostics, and supported by expert system heuristics to ensure accurate results.



1.1.2 APU Neural Net Tool Operational Description

The APU Neural Net Tool (ANNT) resides on the SDC Advanced Application Server. To access ANNT, a web browser is started on a CLCS workstation. The browser is navigated to the ANNT URL and the ANNT button is clicked on. The web server sends a PC-Xware script to open a connection to the CLCS Workstation. The web browser on the CLSC Workstation launches PC-Xware and runs the script. The script opens a connection to the SDC Advanced Application Server. The user is prompted for a password for SDC Advanced Application Server. Once user id and password are accepted, ANNT is started on the Advanced Application Server and displayed on the CLCS Workstation (via xhost). The first screen displayed on the CLCS Workstation is a matrix showing available vehicles and historical data. The user selects which data source ANNT will monitor and then the ANNT main screen appears.

1.2 APU Neural Net Tool Specifications

1.2.1 APU Neural Net Groundrules

- ANNT will run on the SDC Advanced Application Server.
- At least one Gensym G2 and NeurOn-Line license must be resident on the SDC Advanced Application Server.
- ANNT print capability will be dependent on available and supported network printers.
- ANNT will not have any on-line help. A hard copy user guide will be available.
- APU Engineers must supply APU flight data needed to train the neural nets.
- ANNT will have access to real time data services.
- ANNT will not run on command & control workstations.
- Data (100Hz) must be received from data services in less than 250 milliseconds from the time it is requested by ANNT.
- The minimum monitor size required for efficient operations is 17". A 20" monitor is preferred.

1.2.2 APU Neural Net Functional Requirements

The functional requirements of the APU Neural Net Tool are the following:

1. Supported Tool Sets
2. Supported Environments
3. Support Data Interfaces
4. Main Display
5. Sub Display
6. Trained Neural Nets
7. Start-Up Algorithm
8. Wave-Detection Algorithm
9. Knowledge Base Rules

1 Supported Tool Sets

1.1 ANNT shall be developed using Gensym G2 and NeurOn-Line software.

1.2 ANNT shall be distributed using to CLCS workstations using NCD's PC-Xware.

2 Supported Environments

1.1 ANNT shall be capable of being evoked from a web browser on a CLCS Support Workstation.

3 Supported Data Interfaces

1.1 ANNT shall use SDS CCMS real-time data.

1.2 ANNT shall use SDC CCMS historical data.

Data Analysis and Presentation Data Sources (Support for Thor)

	CDS CCMS Historical	SDC CCMS Historical	SDC CLCS Historical	SDS CCMS Real-time	SDSí CCMS et al Real-time	SCAN Server CCMS Real-time	SDS CLCS Real-time
Real Time				✓			
History		✓					

4 Main Display

1.1 The main display shall provide all three APU outputs.
1.2 The main display shall have colored indicators designating the state of each APU.
1.3 A legend shall be included on the display to indicate conditions vs. colors.
1.4 A mode indicator shall reside on the main display.
1.5 The main display shall display APU chamber pressure designated by V46P0120A.
1.6 The main display shall display APU chamber pressure designated by V46P0220A.
1.7 The main display shall display APU chamber pressure designated by V46P0320A.
1.8 The main display shall have a button to access each sub display.

5 Sub Display

1.1 Sub display 1 shall display APU chamber pressure designated by V46P0120A.
1.2 Sub display 1 shall display a pulse diagnostic window for V46P0120A.
1.3 Sub display 1 shall display an extrapolation vs. classification chart for V46P0120A.
1.4 Sub display 2 shall display APU chamber pressure designated by V46P0220A.
1.5 Sub display 2 shall display a pulse diagnostic window for V46P0220A.
1.6 Sub display 2 shall display an extrapolation vs. classification chart for V46P0220A.
1.7 Sub display 3 shall display APU chamber pressure designated by V46P0320A.
1.8 Sub display 3 shall display a pulse diagnostic window for V46P0320A.
1.9 Sub display 3 shall display an extrapolation vs. classification chart for V46P0320A.

6 Data Source Selection Matrix Display (DSSM)

1.1 DSSM display shall provide access to testing real time data.
1.2 DSSM display shall provide access to testing historical data.
1.3 DSSM shall provide data for OV-102, OV-103, OV-104, OV-105.

7 Trained Neural Nets

1.1 A neural net shall recognize a Nominal waveshape.
1.2 A neural net shall recognize an Aero Gimballing waveshape.
1.3 A neural net shall recognize an Engine Gimballing waveshape.
1.4 A neural net shall recognize a Broken Valve waveshape.

8 Start-up Algorithm

1.1 The start-up algorithm shall notify the user that APU start-up has occurred.

9 Wave-detection Algorithm

1.1 The wave-detection algorithm shall recognize each individual waveshape.
1.2 The wave-detection algorithm shall window out each waveshape.

10 Knowledge Base Rules

1.1 The knowledge base rules shall detect an Aero Gimballed waveshape.

1.2 The knowledge base rules shall detect an Engine Gimballed waveshape.

1.2.3 APU Neural Net Tool Performance Requirements

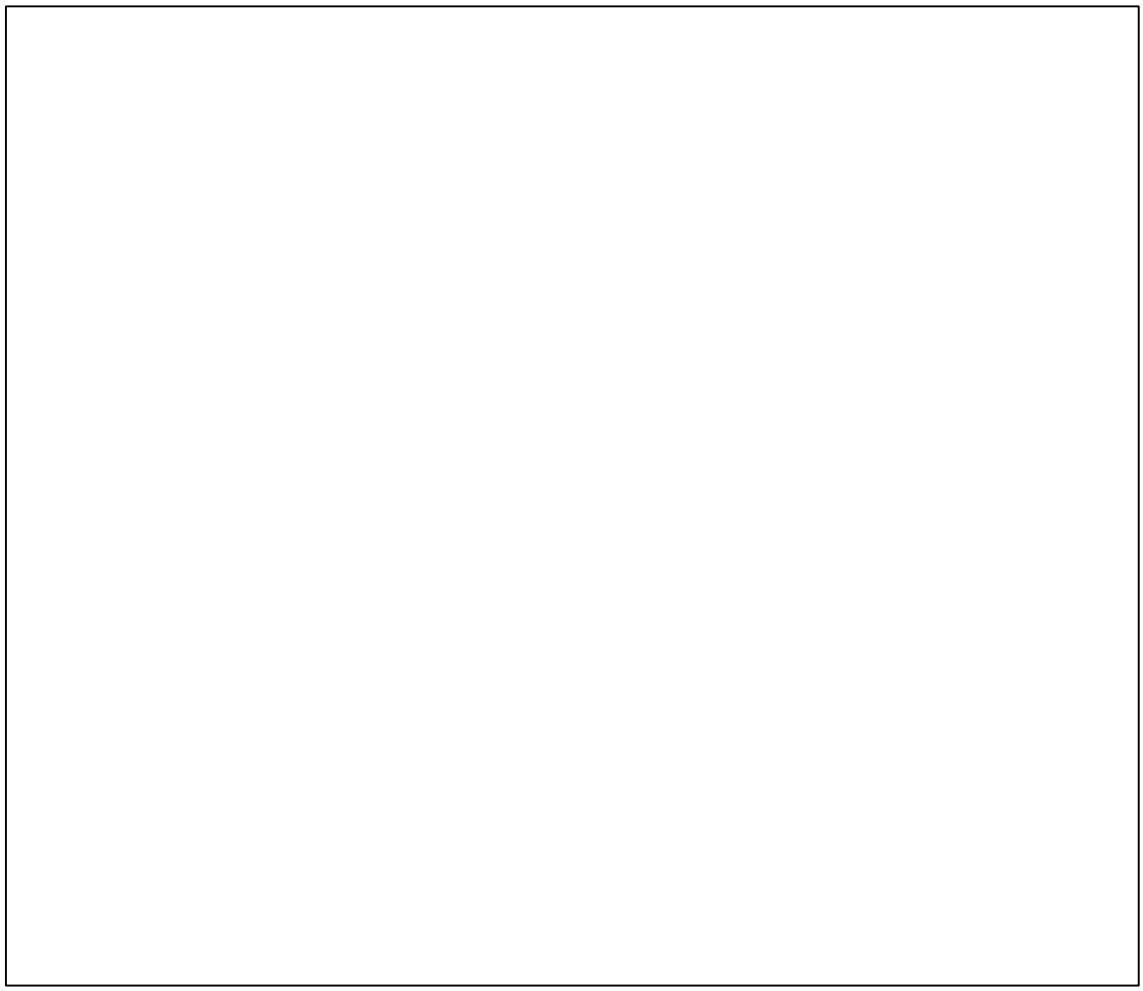
1.1 GSI/ANNT wave detection and windowing must occur in less than 250 milliseconds.

1.2 Main display and neural net classification must occur in less than 500 milliseconds.

1.2.4 APU Neural Net Tool Interface Data Flow Diagrams

At start up, the ANNT Graphic user Interface sends a command to the SDC Advanced Application Server. It starts ANNT/G2 and the G2 Standard Interface, and sends the G2 screen back to the user's workstation. GSI sends the FD list to data services for validation. Every cycle (one second) the GSI sends the current value of FDs into G2. G2 knowledge base accepts the data and processes it for that cycle.

ANNT Data Flow Diagram

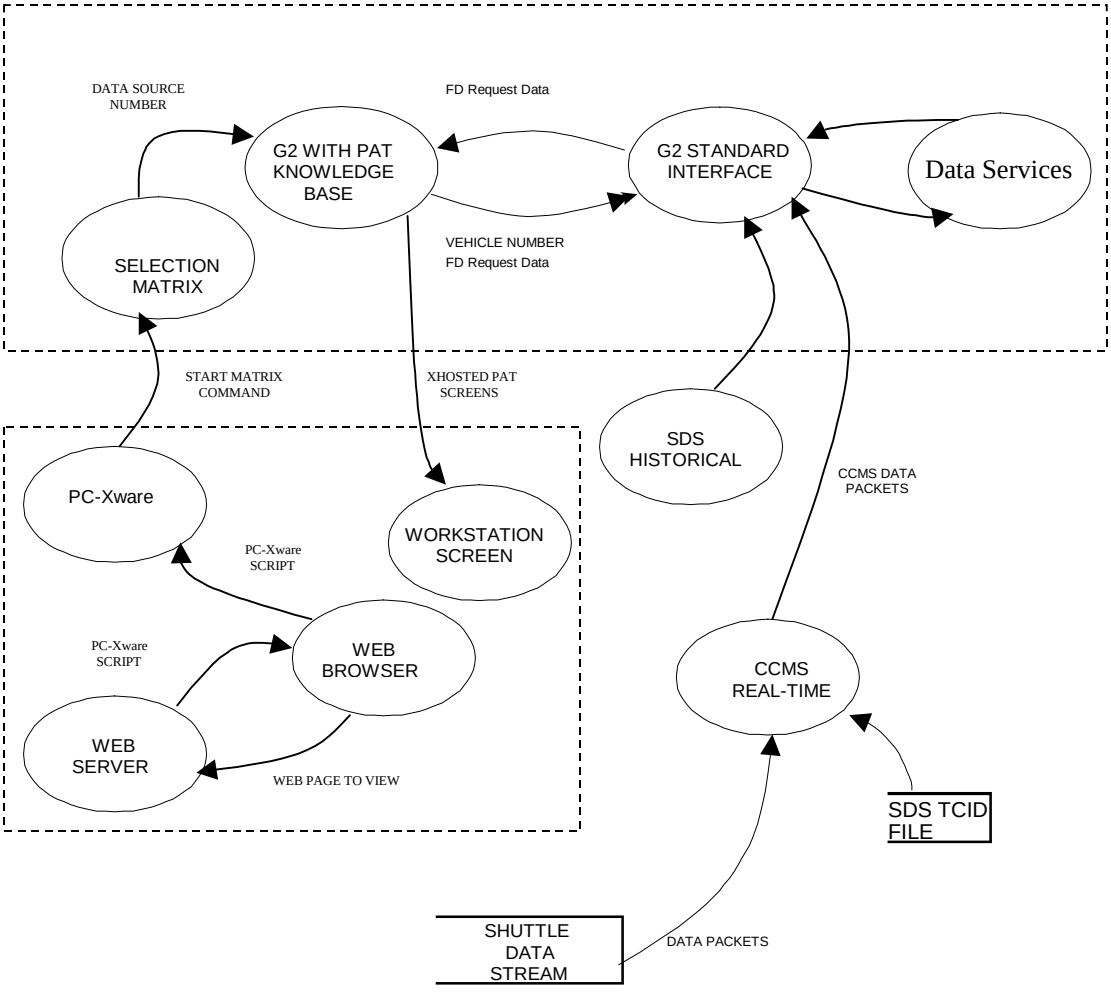


1.3 APU Neural Net Tool Design Specification

APU chamber pressure data is collected from the SDS data stream and placed in shared memory. The GSI reads the shared memory. The gsi performs an algorithm that monitors for the start-up of the APUs. Once the start-up has been recognized, a wave detection algorithm is performed on the next 100-point buffer to extract the chamber pressure wave pulse. The wave pulse data-set and the mode indicator are passed from the gsi interface to G2. G2 passes the wave pulse through the Neural Nets, which classify the data. Ten data points are buffered. The data is queried by the knowledge base rules for the last ten wave pulse classifications and verifies the gimballing type or error mode that is occurring. This state is indicated through multi-color indicators on the real-time display.

1.3.1 APU Neural Net Tool Detailed Data Flow

The user selects a vehicle to have ANNT connect to by clicking the appropriate check box on the ANNT Graphical Users Interface (GUI). Once the user has selected a vehicle, the user clicks the "Start ANNT" button on the ANNT GUI. This sends a command to the SDC Advanced Application Server that tests if the user-selected vehicle is available. If not, an error message is sent to the ANNT GUI telling the user to select another vehicle. Otherwise, ANNT/G2 and ANNT/G2 Standard Interface (ANNT GSI) are started on the server. The ANNT/G2 screen is sent to the user's workstation. ANNT/G2 sends a list of FD's to the ANNT/GSI. The ANNT GSI sends the FD list to data services for validation.



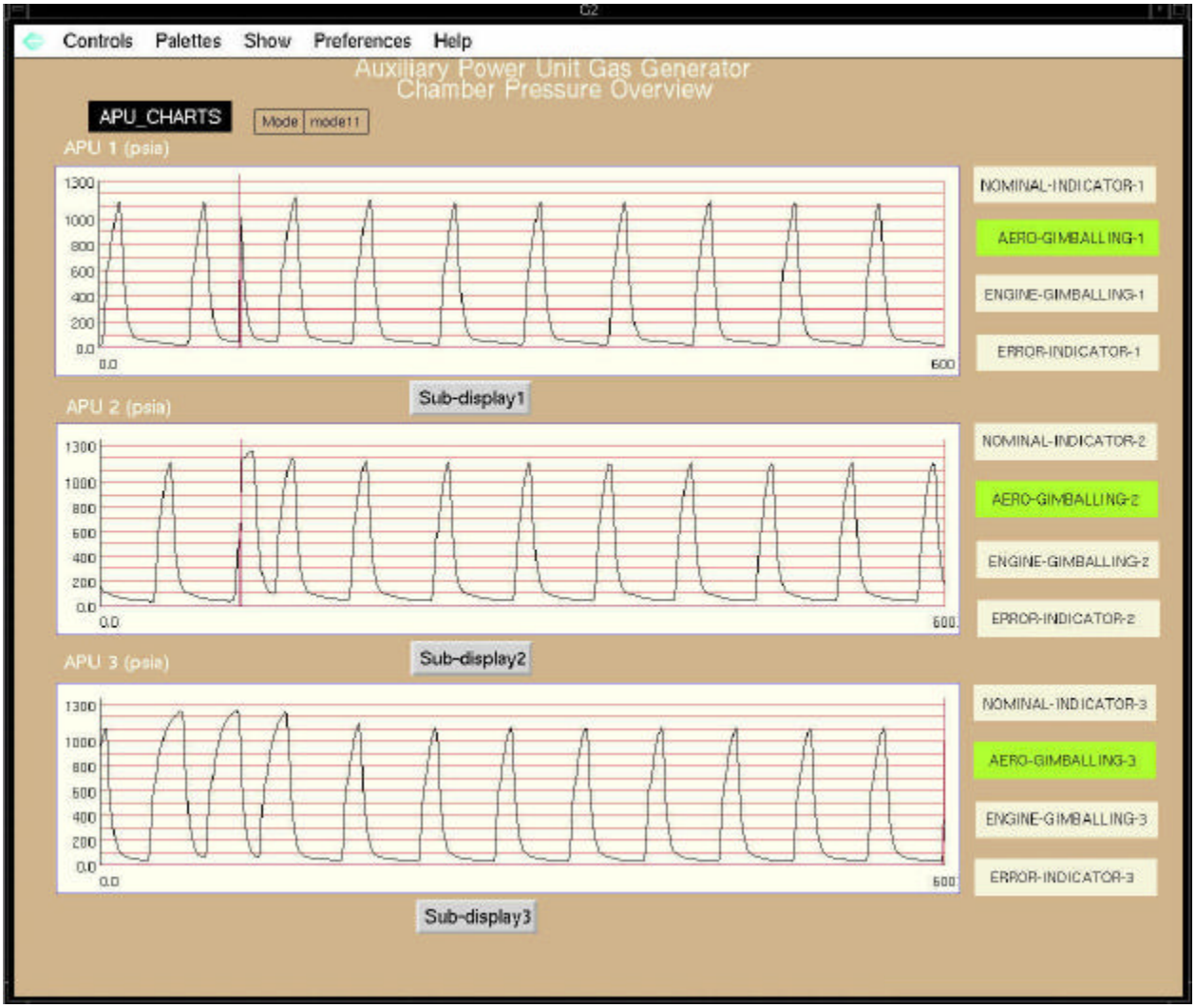
1.3.2 APU Neural Net Tool External Interfaces

1.3.2.1 APU Neural Net Tool Message Formats

Msg#	Description
1000	Mode Indicator – Pre-start, Start-Up, Running, Stopped
1001	The color of the nominal indicator changes from beige to green when a nominal waveshape is recognized by the neural net and returns to beige when the state changes.
1002	The color of the aero-gimballing indicator changes from beige to green when an aero-gimballing is recognized by the neural net and returns to beige when the state changes.
1003	The color of the engine-gimballing indicator changes from beige to green when an engine gimballing is recognized by the neural net and returns to beige when the state changes.
1004	The color of the error indicator changes from beige to indian-red when an error waveshape is recognized by the neural net and returns to beige when the state changes.
1005	APU Neural Net did not receive data from data services in 250 ms from the time it was requested.

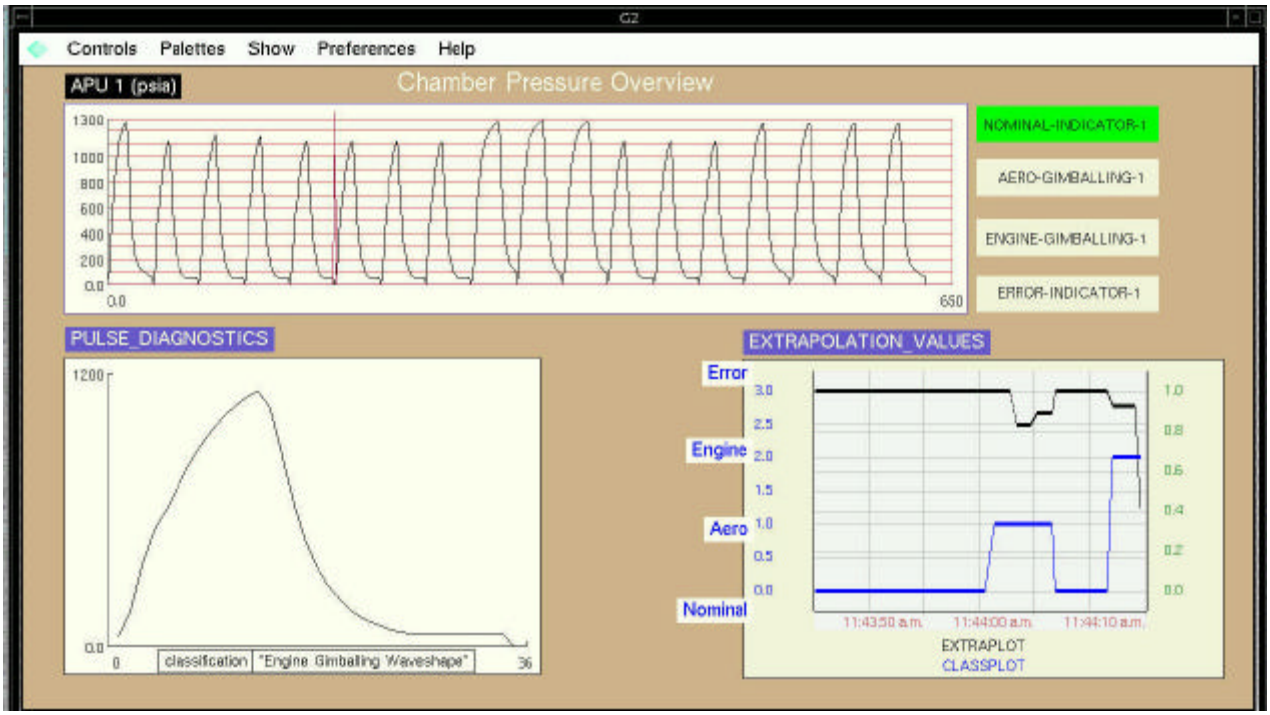
1.3.3 APU Neural Net Tool Display Formats

This is a graphical representation of the ANNT main display.

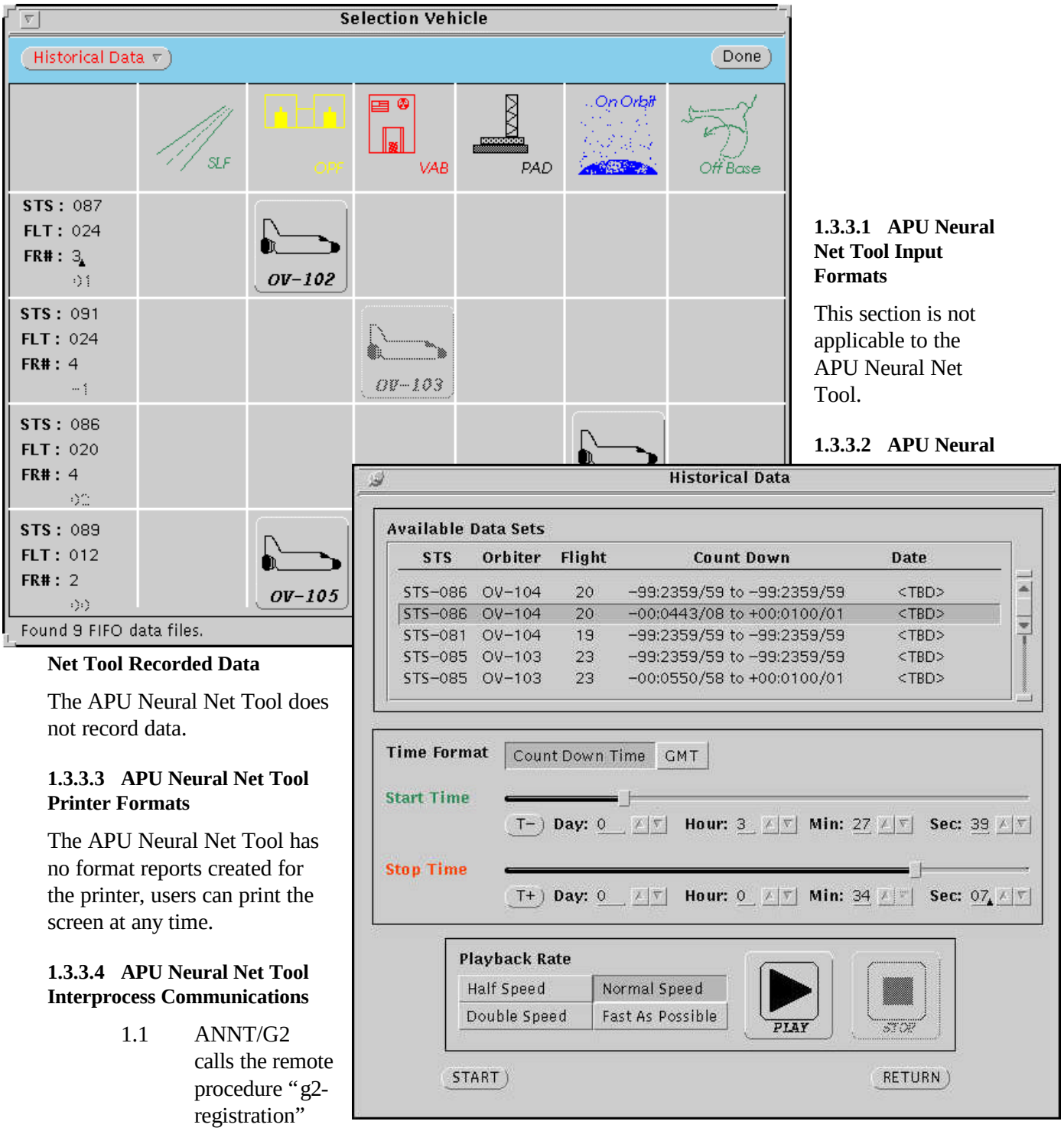


APU Sub Display

This is a graphical representation of the ANNT sub display.


G2
Controls
Palettes
Show
Preferences
Help
APU 1 (psia)
Chamber Pressure Overview
1300
1000
800
600
400
200
0.0
0.0
650
NOMINAL-INDICATOR-1
AERO-GIMBALLING-1
ENGINE-GIMBALLING-1
ERROR-INDICATOR-1
PULSE_DIAGNOSTICS
1200
0.0
0
classification
"Engine Gimballing Waveshape"
36
EXTRAPOLATION_VALUES
Error
Engine
Aero
Nominal
3.0
2.5
2.0
1.5
1.0
0.5
0.0
1.0
0.8
0.6
0.4
0.2
0.0
11:43:50 a.m.
11:44:00 a.m.
11:44:10 a.m.
EXTRAPLOT
CLASSPLOT

This is a graphical representation of the ANNT data source selection matrix display.



1.3.3.1 APU Neural Net Tool Input Formats

This section is not applicable to the APU Neural Net Tool.

1.3.3.2 APU Neural Net Tool Recorded Data

The APU Neural Net Tool does not record data.

1.3.3.3 APU Neural Net Tool Printer Formats

The APU Neural Net Tool has no format reports created for the printer, users can print the screen at any time.

1.3.3.4 APU Neural Net Tool Interprocess Communications

1.1 ANNT/G2 calls the remote procedure "g2-registration" within ANNT/GSI whenever a sensor is first referenced. The "g2_registration" procedure receives the name of the sensor. ANNT/GSI uses this information to build a FD request packet that it will send to data services every second.

See Appendix B. for a detailed description of communications structures.

1.3.3.5 APU Neural Net Tool External Interface Calls (e.g. API calling Formats)

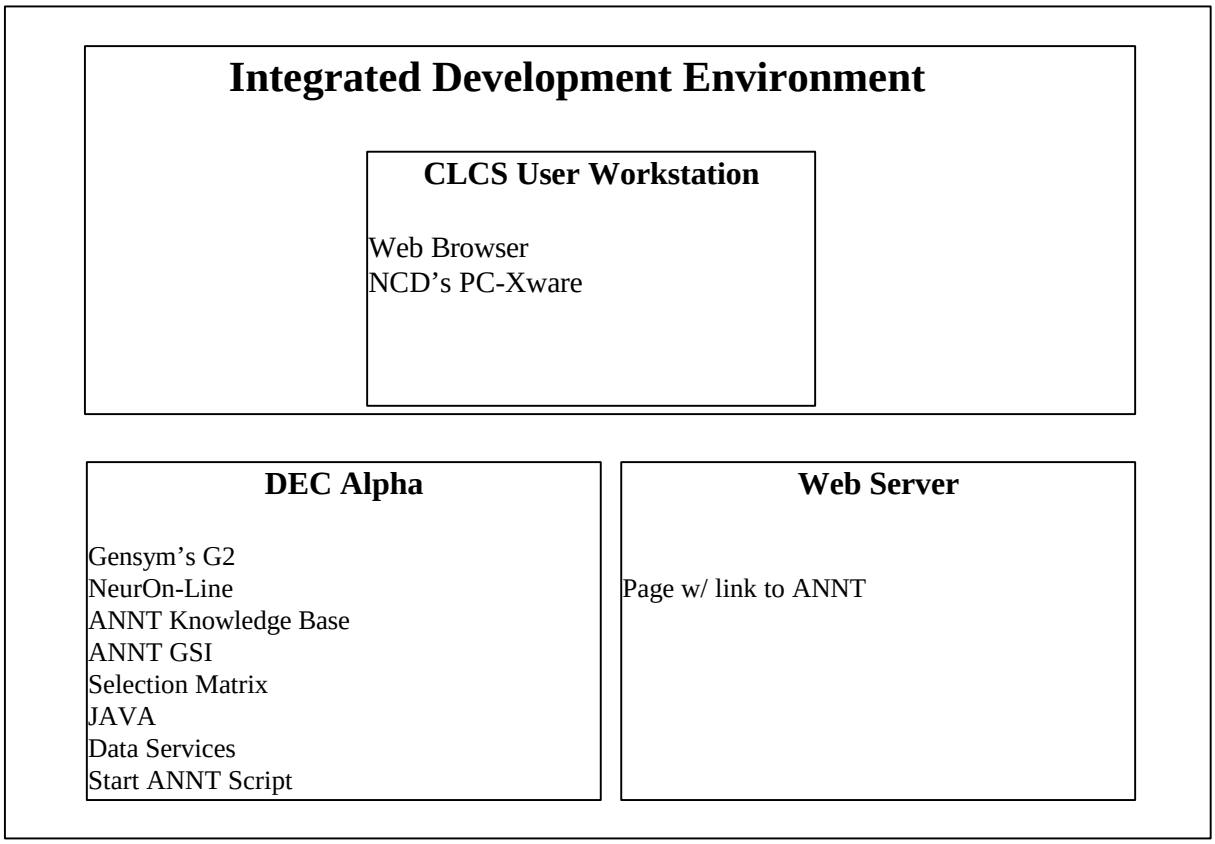
1. APU Neural Net Tool API calls to Data Recording & Distribution.
2. APU Neural Net Tool API calls UCMS data services.

1.3.3.6 APU Neural Net Tool Table Formats

This section is not applicable to APU Neural Net Tool.

1.3.4 APU Neural Net Tool Test Plan

The following is a diagram of the test environment:



1.1 The following hardware is needed for the APU Neural Net Test Plan:

- A CLCS workstation in the Integrated Development Environment.
- A DEC Alpha Advanced Application support workstation in the LCC.
- Web Server with PC-Xware link to ANNT on the Advanced Application Server.

1.2 The following software configuration will be needed on the CLCS workstation:

- Install a web browser (Netscape Navigator 4.0)
- Install NCD's PC-Xware
- Configure web browser to launch PC-Xware application.

1.3 The following software configuration will be needed on the DEC Alpha workstation (Appaloosa):

- Gensym's G2 expert system software

- Gensym's NeurOn-Line software
- The APU Neural Net Tool Knowledge Base
- The APU Neural Net Tool G2 Standard Interface
- Data services that support Data Health
- Script to start ANNT and GSI.

1.4 The following personnel are needed to perform test:

- Quality
- CSCI Lead
- BNA (Boeing North American)

High Level Test Plan Steps:

1. Sign onto the CLCS Workstation and start the web browser.
2. Change web browser to the ANNT URL and click on ANNT button.
3. Enter userid/password to access Advanced Application Server.
4. Once ANNT has finished loading, right click in the background to start ANNT/G2.
5. Main display will appear.
6. Start test file1 that contains nominal data.
7. Review classification results.
8. Start test file2 that contains nominal data and aero gimbal data.
9. Review classification results.
10. Start test file3 that contains nominal data, aero gimbal data, and engine gimbal data.
11. Review classification results.
12. Start test file4 that contains nominal data, aero gimbal data, engine gimbal data, and error data.
13. Review classification results.

Appendix A

	CDS CCMS Historical	SDC CCMS Historical	SDC CLCS Historical	SDS CCMS Real-time	SDS' CCMS et al Real-time	SCAN Server CCMS Real-time	SDS CLCS Real-time
RCWI	✓	✓	* FD only				
ADAT		✓	✓				
RDP		* FD only	✓				
PAT		✓	*	✓	*		*
JView		✓	*	✓	✓	✓	*
ANNT		✓	*	✓	*		*

RCWI - Robust CAP Web Interface
ADAT - Advanced Data Analysis Tool
RDP - Retrieved Data Presentation
PAT - Propulsion Advisory Tool
JView - JView (Java Version of PCGOAL)
ANNT - APU Neural Net Tool

* indicates potential future supported data source.

Appendix B

Structures for G2 to G2 Standard Interface communication.

```
struct info{                        /* record position in file is firing room number */
        char   tcid[10];
        ushort list_number;         /** NTP Area Number                    **/
        ushort type;                /** ? = fifo, ? = ccp                  **/
        ushort orbiter;             /** Orbiter tail number                **/
        ushort tcid_available;      /** Is tcid file in TCID_PATH directory **/
        ushort processed; /** Is NTP processing this. if 0 not processing  **/
};

struct requestSTRUCT
{
   struct srt_req_head              head;
   struct unv_MSID                  parms[NPARMS];
};

struct dbrespSTRUCT                                     /** net_write - verify / hand shaking **/
{
   struct net_header                hdr;
   struct unv_db_desc               desc[NPARMS];
};

struct dataraspSTRUCT                                   /** net_write - send data**/
{
   struct unv_sampdata_hd           head;
   struct unv_samp_data    data[NPARMS];
};

struct dhrespSTRUCT
{
   struct net_header hdr;
   struct srt_data_header
   {
        char dbfil[81]; /* Database file name. */
   } data_header;
};

struct status_packet
{
   char   HostName[32];             /** Host name of Data Server **/
   char   HostIP[32];               /** IP address of Data Server **/
   char   Type[16];                 /** CCP, FIFO, ... **/
   char   TCID[16];

   int          Available;          /** was FiringRoom;   **/
   int          Area;               /** share memory area **/
   int          OrbiterNumber;
   int          STS;
   int          CountDown;          /** seconds **/
   int          GMT;                /** seconds or is it milli-seconds **/
   int          CurrentStatus;      /** see above **/
```

```
   int               NetworkStatus;              /** see above **/
   int               VehicleLocation; /** see above **/
};

struct statusSTRUCT
{
   struct srt_req_head                 head;
   struct status_packet      status[ 25 ];
};
```